Performance Report for:

https://mccordgh.github.com/

Report generated: Tue, Feb 14, 2017, 9:56 AM -0600
Test Server Region: Dallas, USA
Using: Chrome (Desktop) 54.0.2840.71, PageSpeed 1.15-gt1, YSlow 3.1.8

PageSpeed Score	YSlow Score	Fully Loaded Time	Total Page Size	Requests
F (36%)	C (77%)	0.9s	3.46MB	23

Top 5 Priority Issues

Serve scaled images	F (0)	AVG SCORE: 71%	IMAGES	HIGH
Optimize images	F (28)	AVG SCORE: 68%	IMAGES	HIGH
Leverage browser caching	F (42)	AVG SCORE: 59%	SERVER	HIGH
Avoid landing page redirects	F (45)	AVG SCORE: 99%	SERVER	HIGH
Inline small JavaScript	A (92)	AVG SCORE: 95%	JS	HIGH

How does this affect me?

Studies show that users leave a site if it hasn't loaded in 4 seconds; keep your users happy and engaged by providing a fast performing website.

As if you didn't need more incentive, **Google has announced that they are using page speed in their ranking algorithm.**

About GTmetrix

We can help you develop a faster, more efficient, and all-around improved website experience for your users. We use Google PageSpeed and Yahoo! YSlow to grade your site's performance and provide actionable recommendations to fix these issues.

About the Developer

GT.net Performance Hosting

GTmetrix is developed by the good folks at **GT.net**, a Vancouver-based performance hosting company with over 21 years experience in web technology.

https://gt.net/

What do these grades mean?

This report is an analysis of your site with Google and Yahoo!'s metrics for how to best develop a site for optimized speed. The **grades you see represent** how well the scanned URL adheres to those rules.

Lower grades (C or lower) mean that the page can stand to be faster using better practices and optimizing your settings.

What's in this report?

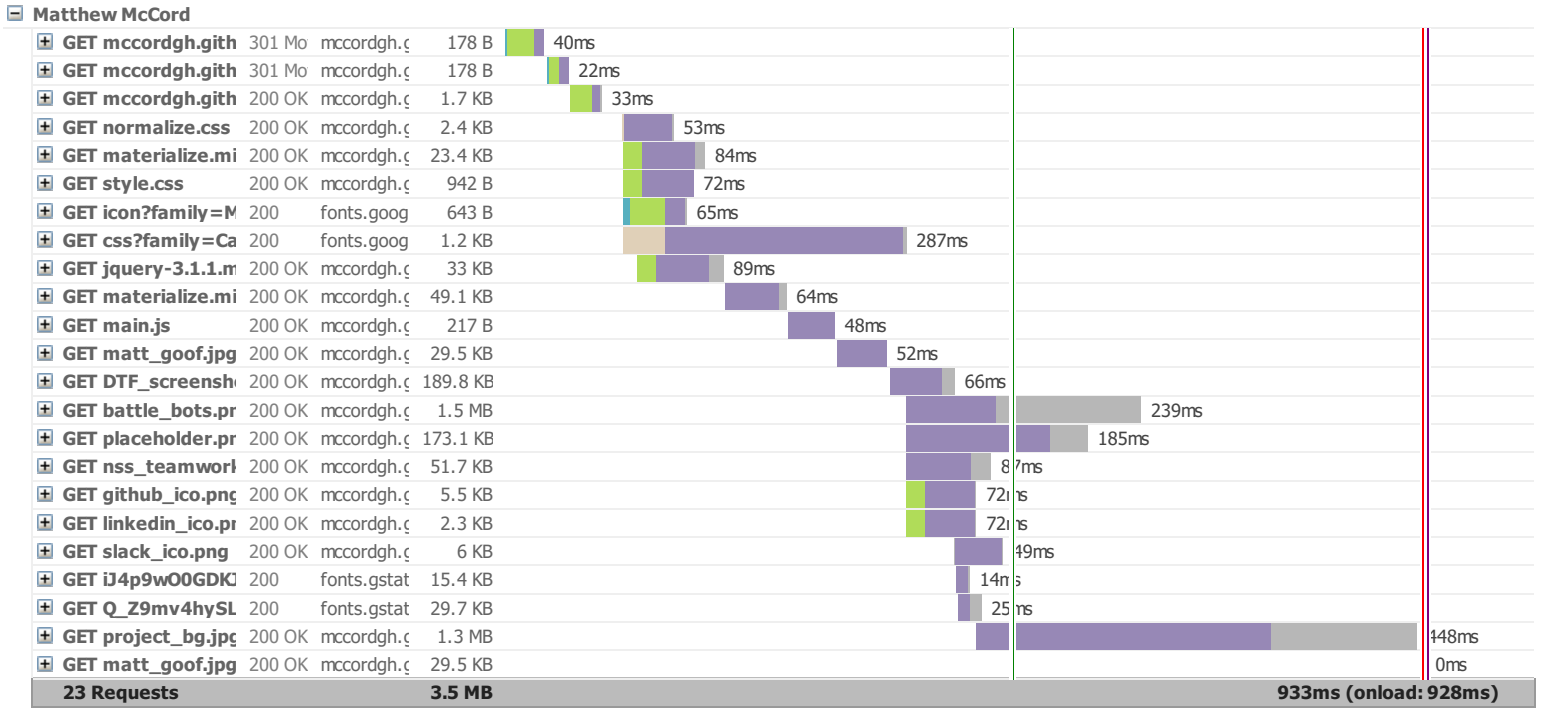
This report covers basic to technical analyses on your page. It is categorized under many headings:

- **Executive:** Overall score information and Priority Issues
- **History:** Graphed history of past performance
- **Waterfall:** Graph of your site's loading timeline
- **Technical:** In-depth PageSpeed & YSlow information

These will provide you with a snapshot of your performance.

Waterfall Chart

The waterfall chart displays the loading behaviour of your site in your selected browser. It can be used to discover simple issues such as 404's or more complex issues such as external resources blocking page rendering.

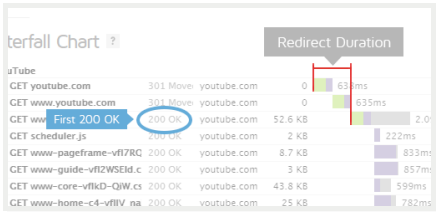


Page Load Timings

RUM Speed Index: 928

66ms Redirect	22ms Connect	10ms Backend	98ms TTFB
0.5s DOM int.	0.5s (1ms) DOM loaded	0.5s First paint	0.9s (0ms) Onload

Redirect duration



This is the time spent redirecting URLs before the final HTML page is loaded. Common redirects include:

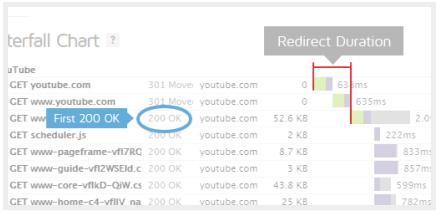
- Redirect from a non-www to www (eg. example.com to www.example.com)
- Redirect to a secure URL (eg. http:// to https://)
- Redirect to set cookies
- Redirect to a mobile version of the site

Some sites may even perform a chain of multiple redirects (eg. non-www to www, then to a secure URL). This timing is the total of all this time that's spent redirecting, or 0 if no redirects occurred.

In the Waterfall Chart, Redirect duration consists of the time from the beginning of the test until just before we start the request of the final HTML page (when we receive the first 200 OK response).

During this time, the browser screen is blank! Ensure that this duration is kept to short by minimizing your redirects.

Connection duration



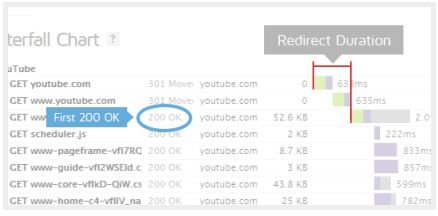
Once any redirects have completed, Connection duration is measured. This is the time spent connecting to the server to make the request to the page.

Technically speaking, this duration is a combination of the blocked time, DNS time, connect time and sending time of the request (rather than *just* connect time). We've combined those components into a single Connection duration to simplify things (as most of these times are usually small).

In the Waterfall Chart, Connection duration consists of everything up to and including the "Sending" time in the final HTML page request (the first 200 OK response).

During this time, the browser screen is still blank! Various causes could contribute to this, including a slow/problematic connection between GTmetrix and the server or slow response times from the server.

Backend duration

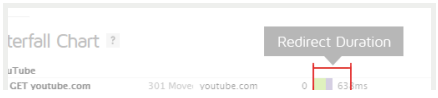


Once the connection is complete and the request is made, the server needs to generate a response for the page. The time it takes to generate the response is known as the Backend duration.

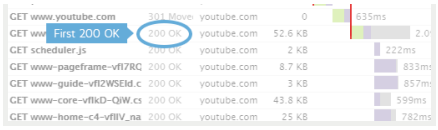
In the Waterfall Chart, Backend duration consists of purple waiting time in the page request.

There are a number of reasons why Backend duration could be slow. We cover this is our "Why is my page slow" article.

Time to First Byte (TTFB)



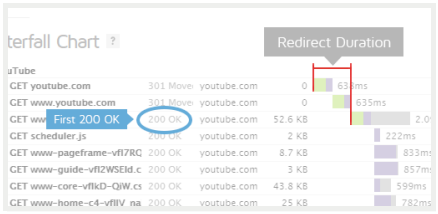
Time to First Byte (TTFB) is the total amount of time spent to receive the first byte of the response once it has been requested. It is the sum of "Redirect duration" + "Connection duration" + "Backend duration". This metric is one of the key indicators of web performance.



In the Waterfall Chart, it is calculated at the start of the test until just before receiving on the page request and represented by the orange line.

Some ways to improve the TTFB include: optimizing application code, implementing caching, fine-tuning your web server configuration, or upgrading server hardware.

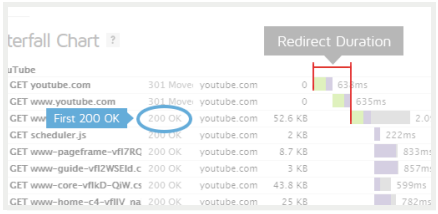
DOM interactive time



DOM interactive time is the point at which the browser has finished loading and parsing HTML, and the DOM (Document Object Model) has been built. The DOM is how the browser internally structures the HTML so that it can render it.

DOM interactive time isn't marked in the Waterfall Chart as it's usually very close in timing to DOM content loaded.

DOM content loaded time



DOM content loaded time (DOM loaded or DOM ready for short) is the point at which the DOM is ready (ie. DOM interactive) and there are no stylesheets blocking JavaScript execution.

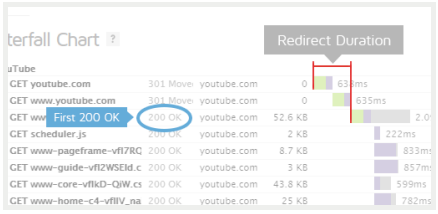
If there are no stylesheets blocking JavaScript execution and there is no parser blocking JavaScript, then this will be the same as DOM interactive time.

In the Waterfall Chart, it is represented by the blue line.

The time in brackets is the time spent executing JavaScript triggered by the DOM content loaded event. Many JavaScript frameworks use this event as a starting point to begin execution of their code.

Since this event is often used by JavaScript as the starting point and delays in this event mean delays in rendering, it's important to make sure that style and script order is optimized and that parsing of JavaScript is deferred.

First paint time

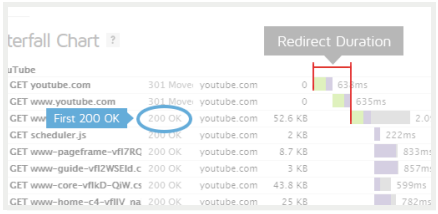


First paint time is the first point at which the browser does any sort of rendering on the page. Depending on the structure of the page, this first paint could just be displaying the background colour (including white), or it could be a majority of the page being rendered.

In the Waterfall Chart, it is represented by the green line.

This timing is of significance because until this point, the browser will have only shown a blank page and this change gives the user an indication that the page is loading. However, we don't know how much of the page was rendered with this paint, so having a early first paint doesn't necessarily indicate a fast loading page.

Onload time



Onload time occurs when the processing of the page is complete and all the resources on the page (images, CSS, etc.) have finished downloading. This is also the same time that DOM complete occurs and the JavaScript window.onload event fires.

Note that there may be JavaScript that initiates subsequent requests for more resources, hence the reason why Fully loaded timing is preferred.

In the Waterfall Chart, it is represented by the red line.

The time in brackets is the time spent executing JavaScript triggered by the Onload event.

Note that Onload time was the previous default for when to stop the test for GTmetrix.

PageSpeed Recommendations

RECOMMENDATION	GRADE	RELATIVE	TYPE	PRIORITY
Serve scaled images	F (0)	AVG SCORE: 71%	IMAGES	HIGH

The following images are resized in HTML or CSS. Serving scaled images could save 1.7MiB (96% reduction).

- https://mccordgh.github.io/img/battle_bots.png is resized in HTML or CSS from 1920x988 to 280x200. Serving a scaled image could save 1.5MiB (97% reduction).
- https://mccordgh.github.io/img/DTF_screenshot.png is resized in HTML or CSS from 1270x798 to 280x200. Serving a scaled image could save 179.4KiB (94% reduction).
- https://mccordgh.github.io/img/nss_teamwork_award.png is resized in HTML or CSS from 993x768 to 200x155. Serving a scaled image could save 49.6KiB (95% reduction).
- https://mccordgh.github.io/img/slack_ico.png is resized in HTML or CSS from 128x128 to 28x32. Serving a scaled image could save 5.7KiB (94% reduction).
- https://mccordgh.github.io/img/github_ico.png is resized in HTML or CSS from 128x128 to 28x32. Serving a scaled image could save 5.2KiB (94% reduction).
- https://mccordgh.github.io/img/linkedin_ico.png is resized in HTML or CSS from 128x128 to 28x32. Serving a scaled image could save 2.2KiB (94% reduction).

Optimize images	F (28)	AVG SCORE: 68%	IMAGES	HIGH

Optimize the following images to reduce their size by 93.4KiB (6% reduction).

- Losslessly compressing https://mccordgh.github.io/img/nss_teamwork_award.png could save 27.7KiB (53% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/DTF_screenshot.png could save 24.5KiB (13% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/placeholder.png could save 22.7KiB (14% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/project_bg.jpg could save 11.2KiB (1% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/slack_ico.png could save 3.0KiB (50% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/github_ico.png could save 2.4KiB (45% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/matt_goof.jpg could save 1.1KiB (4% reduction). See optimized version.
- Losslessly compressing https://mccordgh.github.io/img/linkedin_ico.png could save 634B (27% reduction). See optimized version.

Leverage browser caching	F (42)	AVG SCORE: 59%	SERVER	HIGH

Leverage browser caching for the following cacheable resources:

- https://mccordgh.github.io/css/materialize.min.css (10 minutes)
- https://mccordgh.github.io/css/normalize.css (10 minutes)
- https://mccordgh.github.io/css/style.css (10 minutes)
- https://mccordgh.github.io/img/DTF_screenshot.png (10 minutes)
- https://mccordgh.github.io/img/battle_bots.png (10 minutes)
- https://mccordgh.github.io/img/github_ico.png (10 minutes)
- https://mccordgh.github.io/img/linkedin_ico.png (10 minutes)
- https://mccordgh.github.io/img/matt_goof.jpg (10 minutes)
- https://mccordgh.github.io/img/nss_teamwork_award.png (10 minutes)
- https://mccordgh.github.io/img/placeholder.png (10 minutes)
- https://mccordgh.github.io/img/project_bg.jpg (10 minutes)
- https://mccordgh.github.io/img/slack_ico.png (10 minutes)
- https://mccordgh.github.io/js/jquery-3.1.1.min.js (10 minutes)
- https://mccordgh.github.io/js/main.js (10 minutes)
- https://mccordgh.github.io/js/materialize.min.js (10 minutes)

Avoid landing page redirects	F (45)	AVG SCORE: 99%	SERVER	HIGH

Avoid landing page redirects for the following chain of redirected URLs.

- https://mccordgh.github.com/
- http://mccordgh.github.io/
- https://mccordgh.github.io/

Inline small JavaScript

A (92) | AVG SCORE: 95% | JS | HIGH

The following external resources have small response bodies. Inlining the response in HTML can reduce blocking of page rendering.

https://mccordgh.github.io/ should inline the following small resources:

- https://mccordgh.github.io/js/main.js

Minify CSS

A (99) | AVG SCORE: 94% | CSS | HIGH

Minify CSS for the following resources to reduce their size by 1.8KiB (8% reduction).

- Minifying https://mccordgh.github.io/css/normalize.css could save 1.2KiB (56% reduction) after compression. See optimized version.
- Minifying https://mccordgh.github.io/css/materialize.min.css could save 397B (2% reduction) after compression. See optimized version.
- Minifying https://mccordgh.github.io/css/style.css could save 96B (12% reduction) after compression. See optimized version.
- Minifying https://fonts.googleapis.com/css?family=Cabin|Inconsolata|Nunito|Nunito+Sans|Pacifico|Quicksand|Rubik|VT323 could save 58B (5% reduction) after compression. See optimized version.
- Minifying https://fonts.googleapis.com/icon?family=Material+Icons could save 25B (7% reduction) after compression. See optimized version.

Specify image dimensions

A (99) | AVG SCORE: 98% | IMAGES | MEDIUM

The following image(s) are missing width and/or height attributes.

- https://mccordgh.github.io/img/nss_teamwork_award.png (Dimensions: 993 x 768)

Minify JavaScript

A (99) | AVG SCORE: 87% | JS | HIGH

Minify JavaScript for the following resources to reduce their size by 184B (1% reduction).

- Minifying https://mccordgh.github.io/js/materialize.min.js could save 115B (1% reduction) after compression. See optimized version.
- Minifying https://mccordgh.github.io/js/jquery-3.1.1.min.js could save 46B (1% reduction) after compression. See optimized version.
- Minifying https://mccordgh.github.io/js/main.js could save 23B (12% reduction) after compression. See optimized version.

Minify HTML

A (99) | AVG SCORE: 98% | CONTENT | LOW

Minify HTML for the following resources to reduce their size by 134B (9% reduction).

- Minifying https://mccordgh.github.io/ could save 134B (9% reduction) after compression. See optimized version.

Avoid bad requests

A (100) | AVG SCORE: 98% | CONTENT | HIGH

You scored 100% on this recommendation - nothing to do here!

Defer parsing of JavaScript

A (100) | AVG SCORE: 70% | JS | HIGH

You scored 100% on this recommendation - nothing to do here!

Enable gzip compression	A (100)	AVG SCORE: 81%	SERVER	HIGH

You scored 100% on this recommendation - nothing to do here!

Enable Keep-Alive	A (100)	AVG SCORE: 93%	SERVER	HIGH

You scored 100% on this recommendation - nothing to do here!

Inline small CSS	A (100)	AVG SCORE: 95%	CSS	HIGH

You scored 100% on this recommendation - nothing to do here!

Minimize redirects	A (100)	AVG SCORE: 87%	CONTENT	HIGH

You scored 100% on this recommendation - nothing to do here!

Minimize request size	A (100)	AVG SCORE: 97%	CONTENT	HIGH

You scored 100% on this recommendation - nothing to do here!

Optimize the order of styles and scripts	A (100)	AVG SCORE: 96%	CSS/JS	HIGH

You scored 100% on this recommendation - nothing to do here!

Put CSS in the document head	A (100)	AVG SCORE: 100%	CSS	HIGH

You scored 100% on this recommendation - nothing to do here!

Serve resources from a consistent URL	A (100)	AVG SCORE: 87%	CONTENT	HIGH

You scored 100% on this recommendation - nothing to do here!

Specify a cache validator	A (100)	AVG SCORE: 94%	SERVER	HIGH

You scored 100% on this recommendation - nothing to do here!

Combine images using CSS sprites	A (100)	AVG SCORE: 86%	IMAGES	HIGH

You scored 100% on this recommendation - nothing to do here!

Avoid CSS @import	A (100)	AVG SCORE: 98%	CSS	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Prefer asynchronous resources	A (100)	AVG SCORE: 99%	JS	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Specify a character set early	A (100)	AVG SCORE: 100%	CONTENT	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Avoid a character set in the meta tag	A (100)	AVG SCORE: 100%	CONTENT	LOW

You scored 100% on this recommendation - nothing to do here!

Remove query strings from static resources	A (100)	AVG SCORE: 89%	CONTENT	LOW

You scored 100% on this recommendation - nothing to do here!

Specify a Vary: Accept-Encoding header	A (100)	AVG SCORE: 96%	SERVER	LOW

You scored 100% on this recommendation - nothing to do here!

YSlow Recommendations

RECOMMENDATION	GRADE	RELATIVE	TYPE	PRIORITY
Add Expires headers	F (0)	AVG SCORE: 26%	SERVER	HIGH

There are 17 static components without a far-future expiration date.

- https://mccordgh.github.io/css/normalize.css
- https://mccordgh.github.io/css/materialize.min.css
- https://mccordgh.github.io/css/style.css
- https://fonts.googleapis.com/icon?family=Material+Icons
- https://fonts.googleapis.com/css?family=Cabin|Inconsolata|Nunito|Nunito+Sans|Pacifico|Quicksand|Rubik|VT323
- https://mccordgh.github.io/img/matt_goof.jpg
- https://mccordgh.github.io/img/DTF_screenshot.png
- https://mccordgh.github.io/img/battle_bots.png
- https://mccordgh.github.io/img/placeholder.png
- https://mccordgh.github.io/img/nss_teamwork_award.png
- https://mccordgh.github.io/img/github_ico.png
- https://mccordgh.github.io/img/linkedin_ico.png
- https://mccordgh.github.io/img/slack_ico.png
- https://mccordgh.github.io/js/jquery-3.1.1.min.js
- https://mccordgh.github.io/js/materialize.min.js
- https://mccordgh.github.io/js/main.js
- https://mccordgh.github.io/img/project_bg.jpg

Use a Content Delivery Network (CDN)	F (0)	AVG SCORE: 14%	SERVER	MEDIUM

Using a CDN YSlow doesn't recognize? Specify your CDNs in your User Settings.

There are 15 static components that are not on CDN.

- https://mccordgh.github.io/css/normalize.css
- https://mccordgh.github.io/css/materialize.min.css
- https://mccordgh.github.io/css/style.css
- https://mccordgh.github.io/img/matt_goof.jpg
- https://mccordgh.github.io/img/DTF_screenshot.png
- https://mccordgh.github.io/img/battle_bots.png
- https://mccordgh.github.io/img/placeholder.png
- https://mccordgh.github.io/img/nss_teamwork_award.png
- https://mccordgh.github.io/img/github_ico.png
- https://mccordgh.github.io/img/linkedin_ico.png
- https://mccordgh.github.io/img/slack_ico.png
- https://mccordgh.github.io/js/jquery-3.1.1.min.js
- https://mccordgh.github.io/js/materialize.min.js
- https://mccordgh.github.io/js/main.js
- https://mccordgh.github.io/img/project_bg.jpg

Make fewer HTTP requests	B (88)	AVG SCORE: 35%	CONTENT	HIGH

This page has 5 external stylesheets. Try combining them into one.

Avoid URL redirects	B (80)	AVG SCORE: 86%	CONTENT	MEDIUM

There are 2 redirects

- https://mccordgh.github.com/ redirects to http://mccordgh.github.io/
- http://mccordgh.github.io/ redirects to https://mccordgh.github.io/

Compress components with gzip	A (100)	AVG SCORE: 81%	SERVER	HIGH

You scored 100% on this recommendation - nothing to do here!

Minify JavaScript and CSS	A (100)	AVG SCORE: 74%	CSS/JS	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Make AJAX cacheable	A (100)	AVG SCORE: 100%	JS	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Remove duplicate JavaScript and CSS	A (100)	AVG SCORE: 100%	CSS/JS	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Avoid AlphaImageLoader filter	A (100)	AVG SCORE: 98%	CSS	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Avoid HTTP 404 (Not Found) error	A (100)	AVG SCORE: 99%	CONTENT	MEDIUM

You scored 100% on this recommendation - nothing to do here!

Reduce the number of DOM elements	A (100)	AVG SCORE: 93%	CONTENT	LOW

You scored 100% on this recommendation - nothing to do here!

Use cookie-free domains	A (100)	AVG SCORE: 47%	COOKIE	LOW

You scored 100% on this recommendation - nothing to do here!

Use GET for AJAX requests	A (100)	AVG SCORE: 100%	JS	LOW

You scored 100% on this recommendation - nothing to do here!

Avoid CSS expressions	A (100)	AVG SCORE: 99%	CSS	LOW

You scored 100% on this recommendation - nothing to do here!

Reduce DNS lookups	A (100)	AVG SCORE: 69%	CONTENT	LOW

- mccordgh.github.com: 1 component, 0.1K

- mccordgh.github.io: 17 components, 3817.5K (113.4K GZip)
- fonts.googleapis.com: 2 components, 9.2K (1.8K GZip)

Reduce cookie size	A (100)	AVG SCORE: 100%	COOKIE	LOW

You scored 100% on this recommendation - nothing to do here!

Make favicon small and cacheable	A (100)	AVG SCORE: 100%	IMAGES	LOW

You scored 100% on this recommendation - nothing to do here!

Configure entity tags (ETags)	A (100)	AVG SCORE: 83%	SERVER	LOW

You scored 100% on this recommendation - nothing to do here!

Make JavaScript and CSS external	(n/a)		CSS/JS	MEDIUM

Only consider this if your property is a common user home page.